A Datasheet

A.1 Motivation

- **For what purpose was the dataset created?** In order to benchmark easily new methods on HPO and transfer HPO.

- **Who create the dataset and on behalf of which entity?** The authors of the main paper work together on behalf of their respective institution.

- **Who funded the creation of the dataset?** University of Freiburg, University of Hildesheim and Amazon.

A.2 Composition

- **What do the instances that comprises the dataset represent?** They represent evaluations of hyperparameter configurations on diferent tasks.

- **How many instances are there in total?** Around 6.4 million hyperparameter evaluations.

- **Does the dataset contain all possible instances or is it a sample of instances from a larger set?** No, they correspond only to evaluations of supervised classifiers and queried from OpenML platform.

- **What data does each instance consist of?** Each instance consist of a hyperparameter configuration, with their respective values and response.

- **Is there a label or target associated with each instance?** Yes, the associated accuracy (validation accuracy).

- **Is any information missing from individual instances?** No. The missing values have been imputed.

- **Are relationships between individual instances made explicit?** No.

- **Are there recommended data splits?** Yes, we recommend three splits for meta-train, meta-validation and meta-test.

- **Are there any errors, sources of noise or redundancies in the dataset?** Yes, the evaluations rely on third parties that may have commited mistakes in reporting the response or hyperparameter configuration values.

- **Is the dataset self-contained or does it link to or otherwise rely on external resources?** For the creation, it relied on data from OpenML.

- **Does the dataset contain data that might be considered confidential?** No.

- **Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening or might otherwise cause anxiety?** No.

- **Does the dataset relate to people?** No.

- **Does the dataset identify any sub-populations?** No.

- **Is it possible to identify individuals, either directly or indirectly?** No.

- **Does the dataset contain data that might be considered sensitive in any way?** No.

A.3 Collection process

- **How was the data associated with each instance acquired?** We queried the existing runs (evaluations) with tag Verified_Supervised_Classification from OpenML.

- **What mechanisms or procedures were used to collect the data?** We used the Python API for fetching the data.

- **Who was involved in the data collection process?** Only the authors of this paper.

- **Over what time frame was the data collected?** The existing runs until April 15, 2021.

- **Were any ethical review process conducted?** No.

- **Does the dataset relate to people?** No.

- **Did you collect the data from the individuals in question, or obtained it via third parties or other sources?** It was obtained from OpenML website.
- **Was any preprocessing/cleaning/labeling of the data done?** Yes, it is explain in section 5.
- **Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data?** No, but the raw data can be accessed through OpenML platform.
- **Is the software used to preprocess/clean/label the instances available?** No.

A.4 Uses

- **Has the dataset been used for any tasks already?** Yes, we provide examples in section 7.
- **Is there a repository that links to any or all papers or systems that use the dataset?** Yes, it is available in `https://github.com/releaunifreiburg/HPO-B`.
- **What (other) tasks could the dataset be used for?** HPO and transfer HPO.
- **Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?** No.
- **Are there tasks for which the dataset should not be used?** No.

A.5 Distribution

- **Will the dataset be distributed to third parties outside of the entity?** Yes, the dataset is publicly available.
- **How will the dataset be distributed?** Through a link hosted in our repository `https://github.com/releaunifreiburg/HPO-B`.
- **When will the dataset be distributed?** From June 8. 2021.
- **Will the dataset be distributed under a copyright or other intellectual property license?** Yes, license CC-BY.
- **Have any third parties imposed IP-based or other restrictions on the data associated with the instances?** No.
- **Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?** No.

A.6 Maintenance

- **Who will be supporting/hosting/maintaining the dataset?** RELEA group from the University of Freiburg.
- **How can the owner/curator/manager of the dataset be contacted?** Questions can be sent to `pineda@cs.uni-freiburg.de` or submit an issue to the Github repository `https://github.com/releaunifreiburg/HPO-B`.
- **Is there an erratum?** No.
- **Will the dataset be updated?** No.
- **Will older versions of the dataset continue to be supported/hosted/maintained?** As new runs (evaluations) might be available in OpenML, we consider to update it periodically.
- **If other want to extend/augment/build on/contributed to the dataset, is there a mechanism for them to do so?** We might provide this possibility in the future.

B License

C Accessibility

The link to access the meta-dataset and recommendations on how to consume it are given in the following git-hub repository: `https://github.com/releaunifreiburg/HPO-B`.

D Maintenance

We are planning to increase the meta-data set, as new search spaces with a considerable amount of evaluations are available on the platform. We may also add evaluations from third parties that agree to be part of the meta-dataset. For any suggestion or technical inquiry, we recommend to use the issue tracker of our repository.

## E Additional Meta-dataset Statistics

In this section, we present further descriptive statistics of the meta-dataset. The histogram in Figure 7a depicts how many search spaces (out of the 176) contain a specific number of evaluations. At the task level, the histogram shows a similar pattern in Figure 7b. The Figures 7c and 7d show respectively the percentage of search spaces and tasks with *at least* a given number of evaluations. As can be seen, 50% of the tasks have at least 20000 evaluations, and at least 30% contain 40.000 evaluations (out of 1907 tasks).
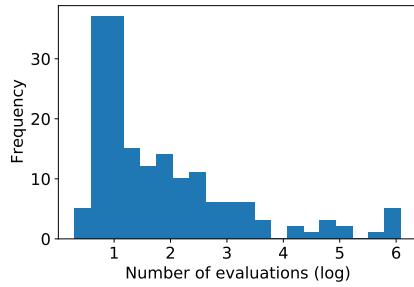
(a) Frequency of evaluations per search space.
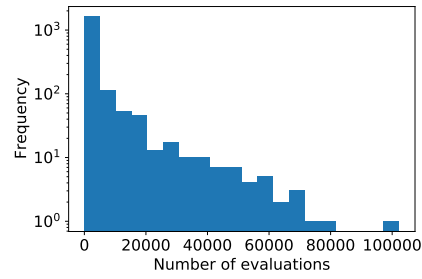
(b) Frequency of the number of evaluations per task.
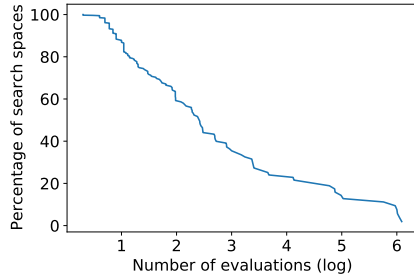
(c) Percentage of search spaces with a given number of evaluations.
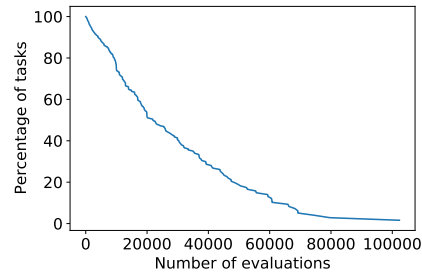
(d) Percentage of search spaces with a given number of evaluations.

Figure 7: Descriptors for the number of evaluations.

On another discourse, the distribution of the response per search-space in the meta-test split can be observed in Figure 8. It is noticeable that the distributions are multi-modal, due to the fact that they correspond to the aggregated response of aggregated tasks. It shows how every task has its own scale. A more-detailed insight of the distribution per dataset is shown in the Figure 9.

F Additional results

Figures 12, 10, and 14 show the results of the average rank, where the ranking was computed per task. Moreover, Figure 13 presents detailed results for the average rank with different acquisition functions that was summarized in the main paper.
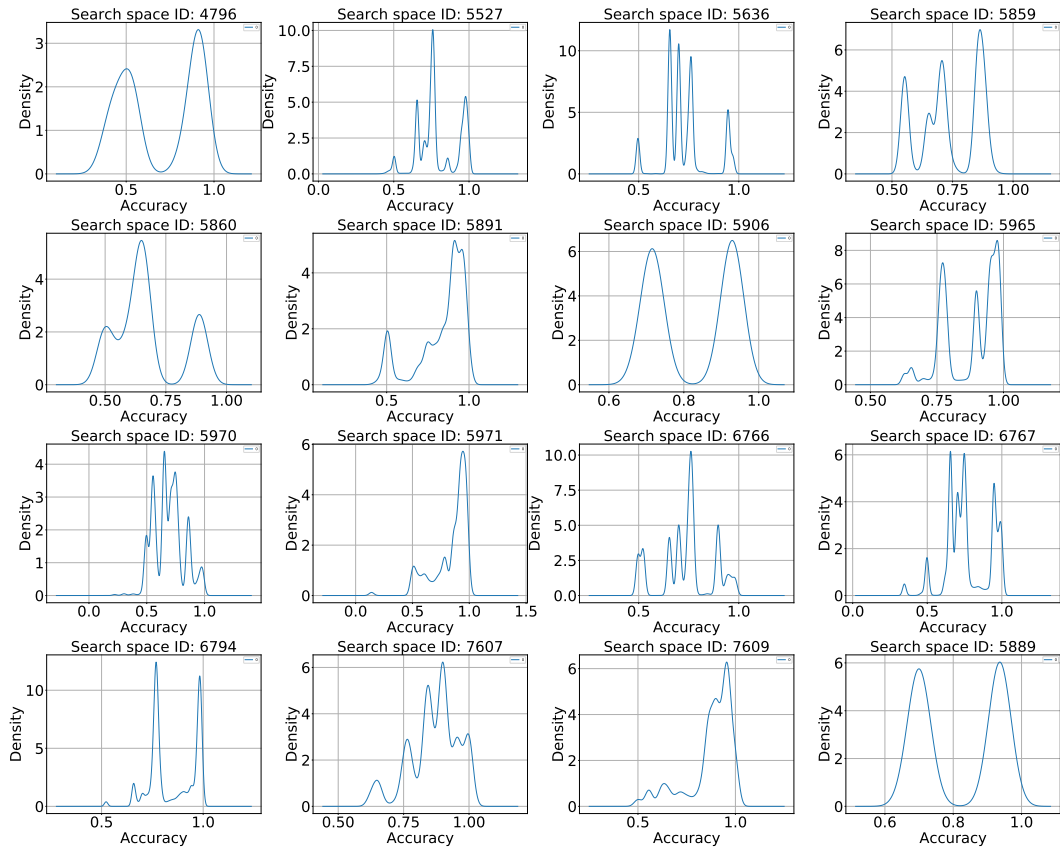
Figure 8: Density distribution of the response per search-space.
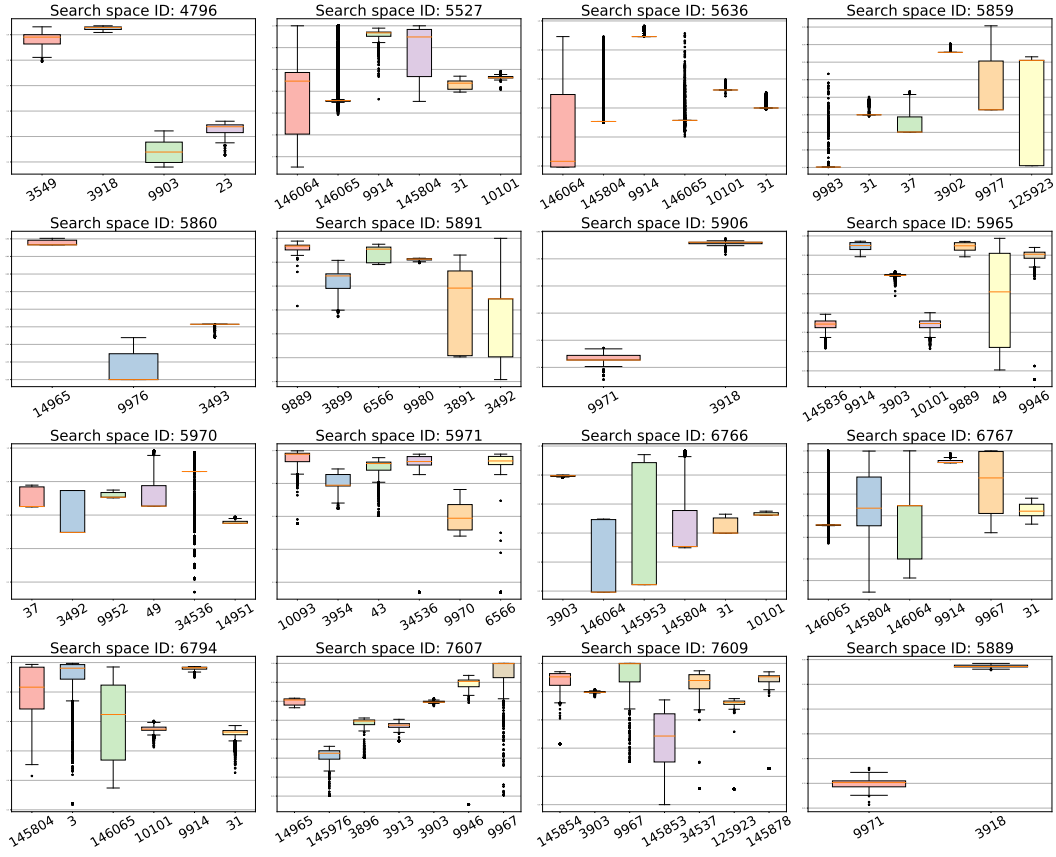
Figure 9: Distribution of the response (accuracy) per task in meta-test.
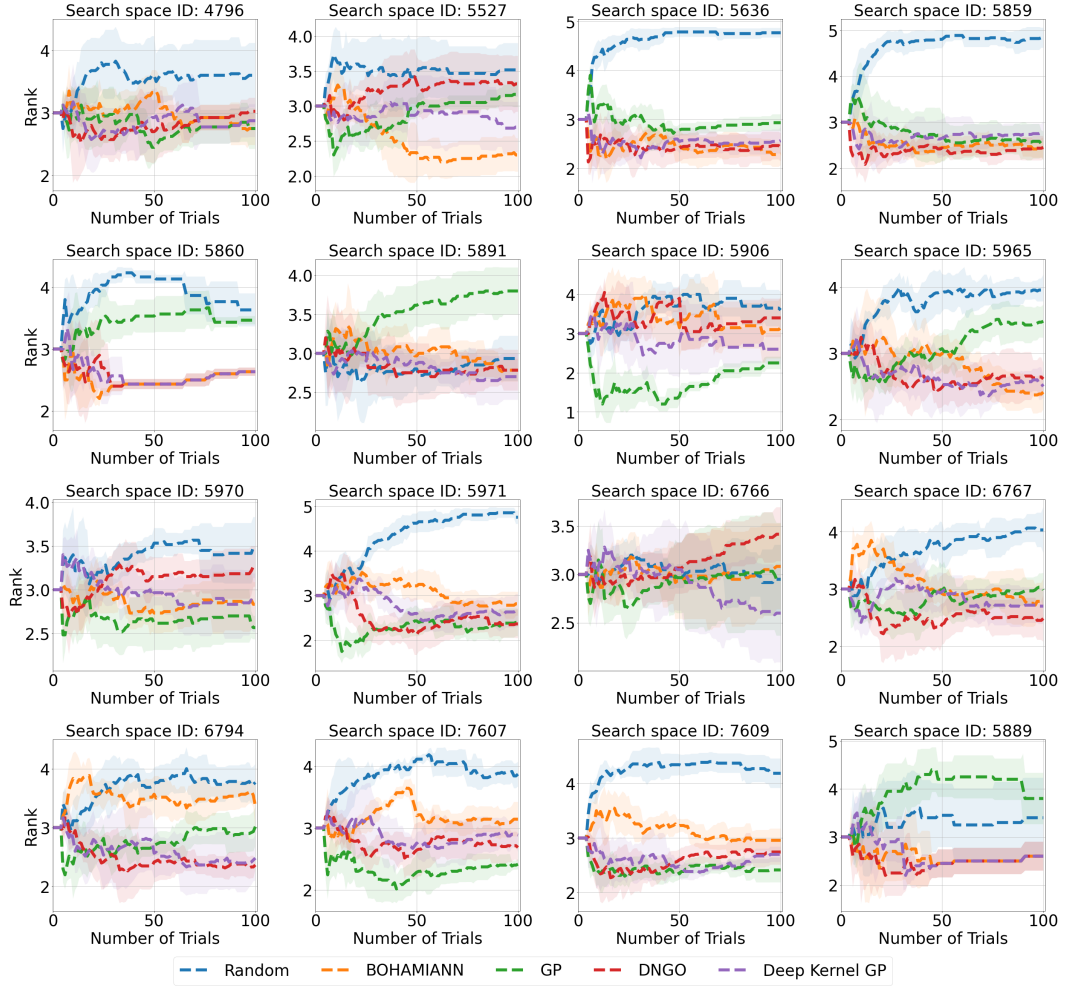
Figure 10: **Mean rank** comparison of **non-transfer** black-box HPO methods
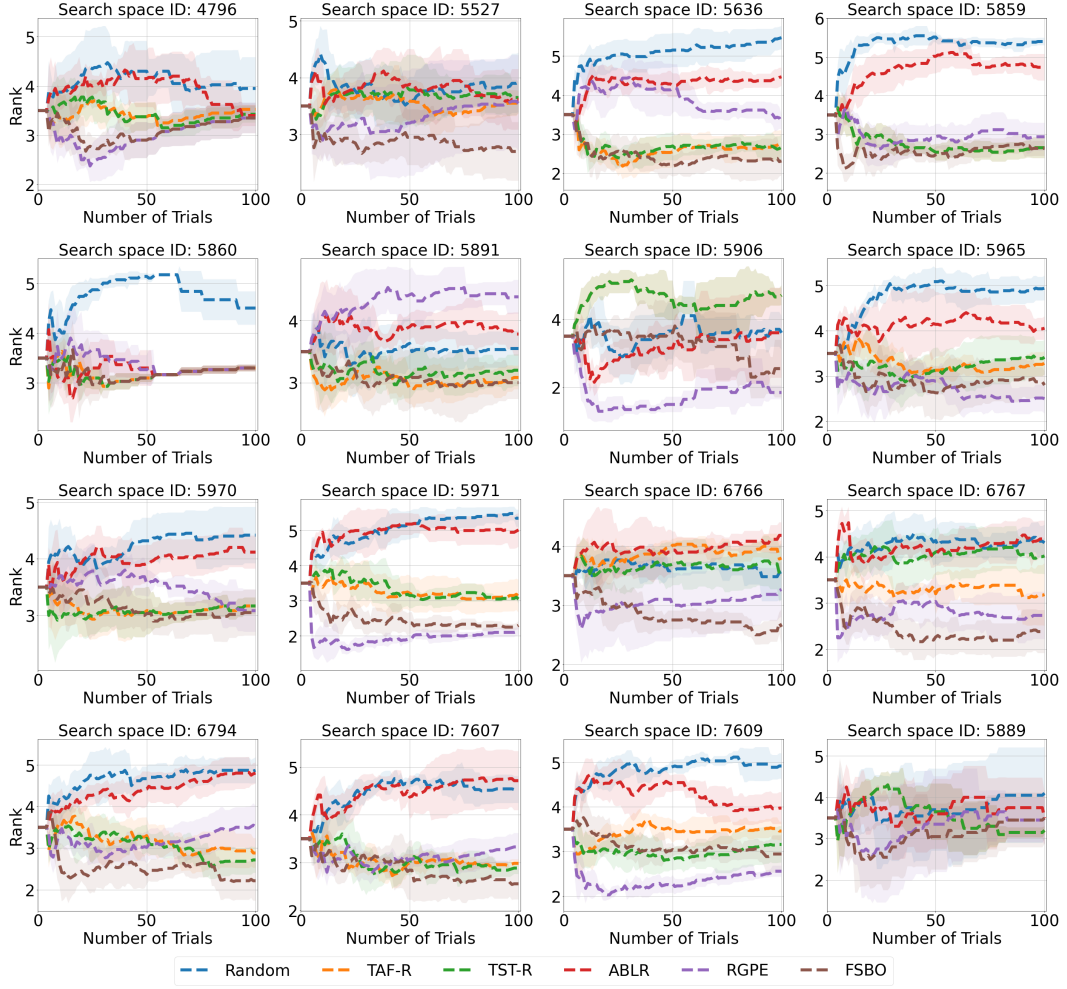
Figure 11: **Mean rank** comparison of **transfer learning** black-box HPO methods
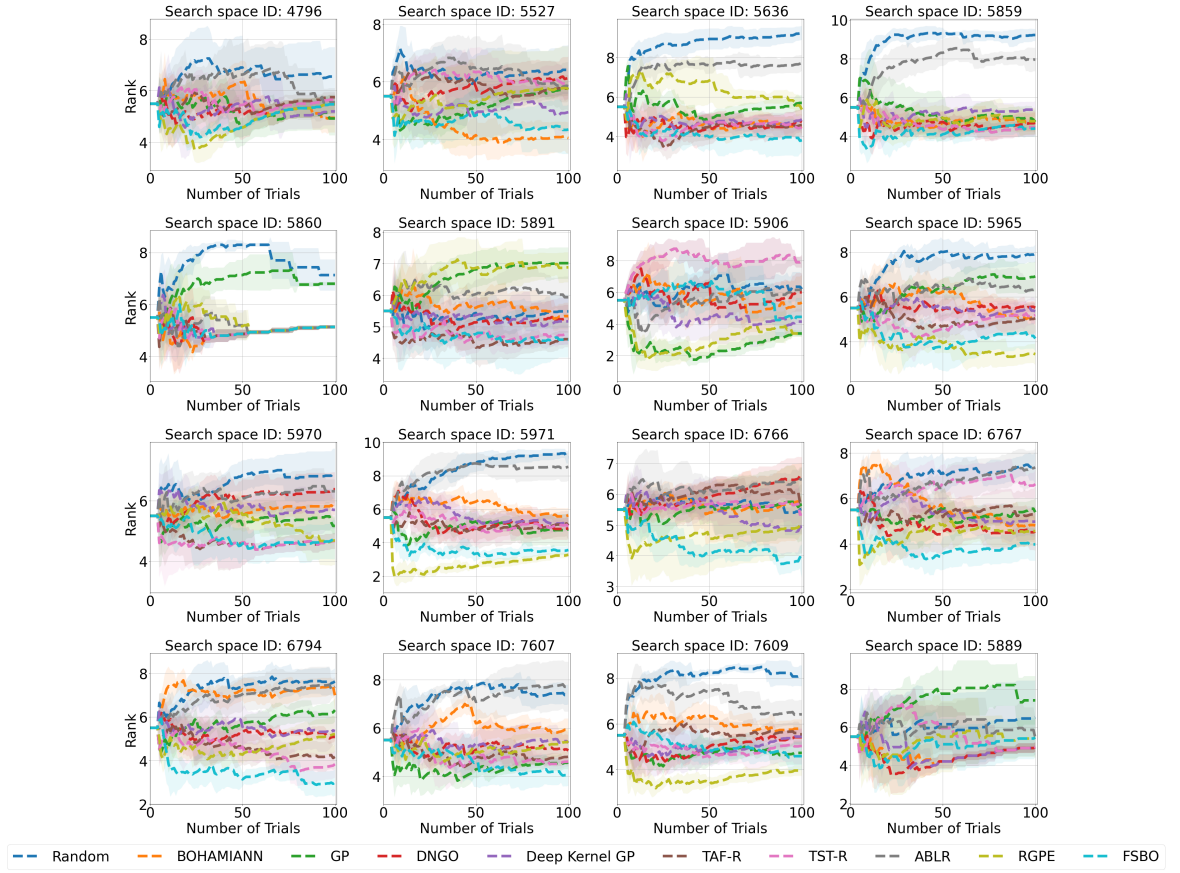
Figure 12: **Mean rank** comparison of **non-transfer** and **transfer learning** black-box HPO methods

Figure 13: **Normalized regret** comparison of different acquisition functions: Expected Improvement (EI), Entropy (E) and Upper Confidence Bound (UCB) for Random Forest (RF) and Boosted Trees (BT) as surrogates.

Figure 14: **Mean rank** comparison of different acquisition functions

Table 4: Hyperparameters of the search spaces HPO-B-v2/-v3

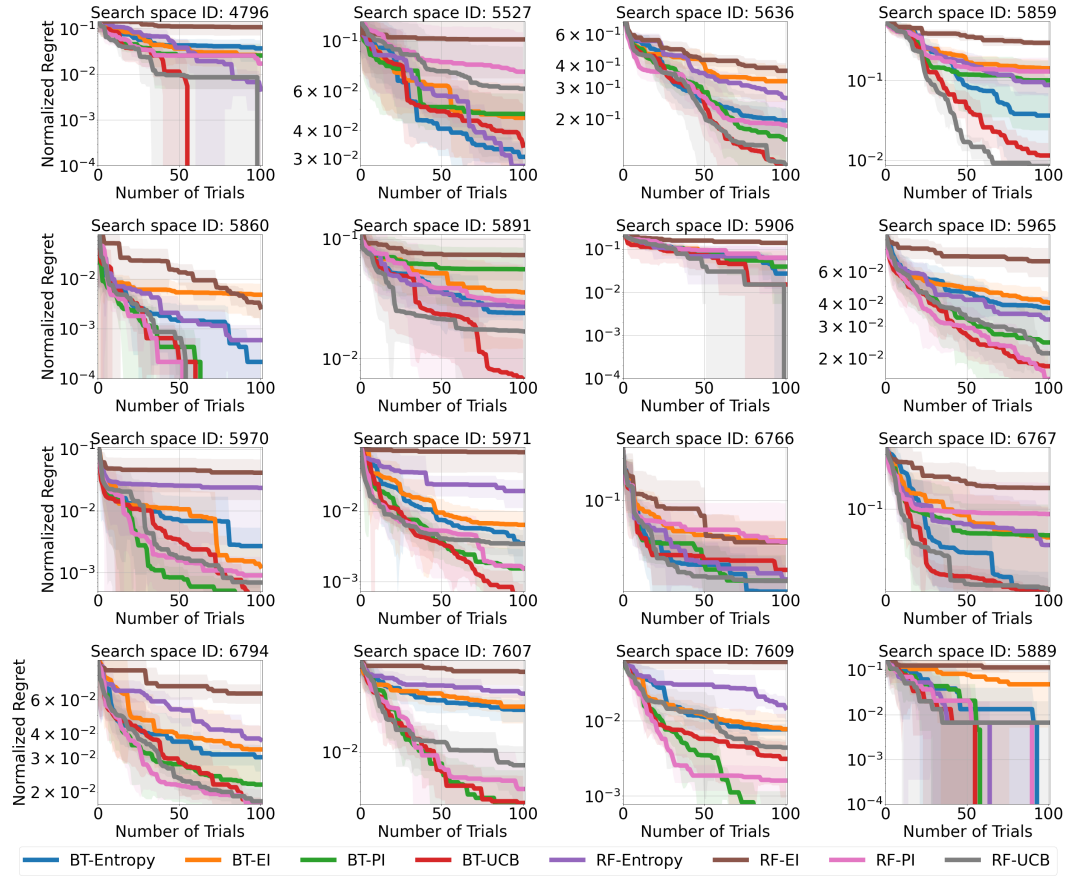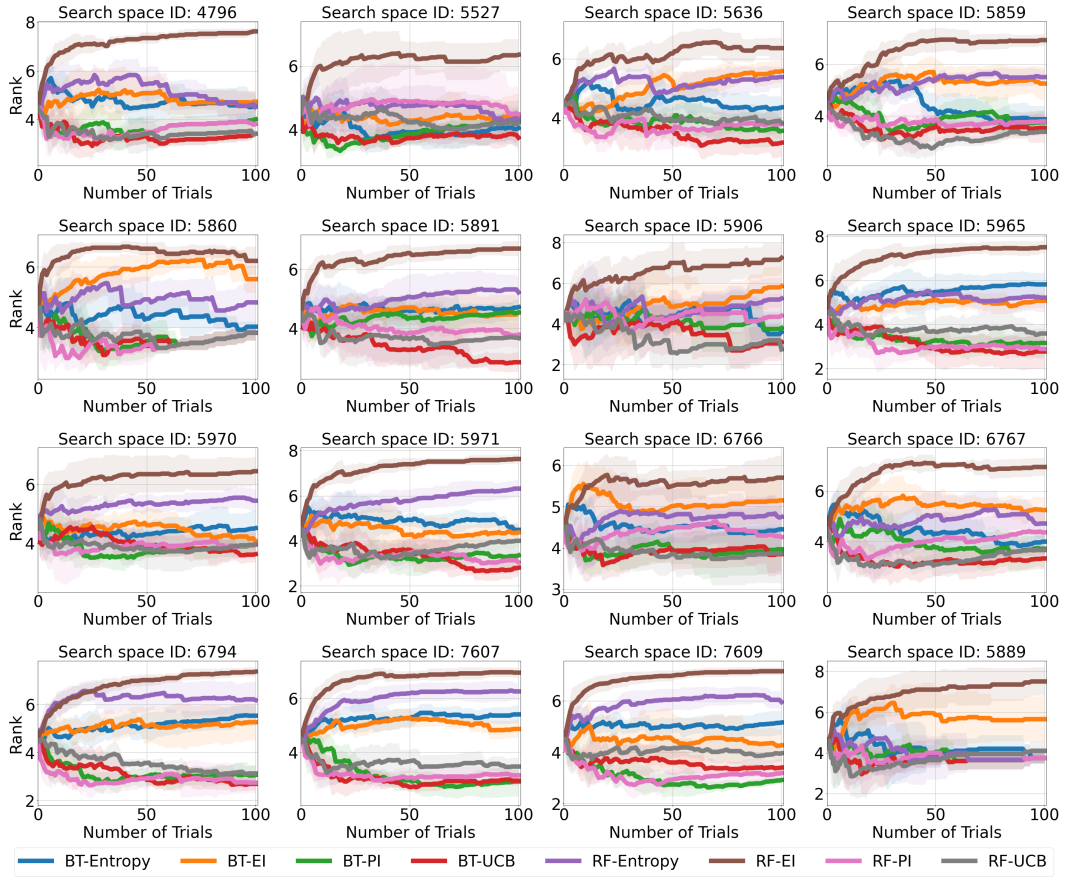| Search Space Id	Search Space Name	List of hyperparameters within the Search-space
4796	mlr.classif.rpart .preproc(16)	minsplit, minbucket, cp
5527	mlr.classif.svm (6)	cost, gamma, gamma.na, degree, degree.na, kernel.ohe.na, kernel.ohe.linear, kernel.ohe.polynomial
5636	mlr.classif.rpart (29)	minsplit, minsplit.na, minbucket, cp, maxdepth, maxdepth.na
5859	mlr.classif.rpart (31)	minsplit, minsplit.na, minbucket, cp, maxdepth, maxdepth.na
5860	mlr.classif.glmnet (4)	alpha, lambda
5891	mlr.classif.svm (7)	cost, gamma, gamma.na, degree, degree.na, kernel.ohe.na, kernel.ohe.linear, kernel.ohe.polynomial
5906	mlr.classif.xgboost (4)	eta, max_depth, max_depth.na, min_child_weight, min_child_weight.na, subsample, colsample_bytree, colsample_bytree.na, colsample_bylevel, colsample_bylevel.na, lambda, alpha, nrounds, nrounds.na, booster.ohe.na, booster.ohe.gblinear
5965	mlr.classif.ranger(9)	num.trees, num.trees.na, mtry, sample.fraction, min.node.size, min.node.size.na, replace.ohe.FALSE, replace.ohe.na, respect.unordered.factors.ohe.INVALID, respect.unordered.factors.ohe.TRUE
5970	mlr.classif.glmnet(5)	alpha, lambda
5971	mlr.classif.xgboost (6)	eta, max_depth, max_depth.na, min_child_weight, min_child_weight.na, subsample, colsample_bytree, colsample_bytree.na, colsample_bylevel, colsample_bylevel.na, lambda, alpha, nrounds, nrounds.na, booster.ohe.na, booster.ohe.gblinear
6766	mlr.classif.glmnet (11)	alpha, lambda
6767	mlr.classif.xgboost (9)	eta, subsample, lambda, alpha, nthread, nthread.na, nrounds, nrounds.na, max_depth, max_depth.na, min_child_weight, min_child_weight.na, colsample_bytree, colsample_bytree.na, colsample_bylevel, colsample_bylevel.na, booster.ohe.na, booster.ohe.gblinear
6794	mlr.classif.ranger(13)	num.trees, num.trees.na, mtry, sample.fraction, min.node.size, min.node.size.na, replace.ohe.FALSE, replace.ohe.na, respect.unordered.factors.ohe.na respect.unordered.factors.ohe.TRUE
7607	mlr.classif.ranger(15)	num.trees, num.trees.na, mtry, min.node.size, sample.fraction, respect.unordered.factors.ohe.na, respect.unordered.factors.ohe.TRUE, replace.ohe.FALSE, replace.ohe.na
7609	mlr.classif.ranger(16)	num.trees, num.trees.na, mtry, min.node.size, sample.fraction, respect.unordered.factors.ohe.na, respect.unordered.factors.ohe.TRUE, replace.ohe.FALSE, replace.ohe.na
5889	mlr.classif.ranger(7)	num.trees, num.trees.na, mtry, sample.fraction, replace.ohe.FALSE, replace.ohe.na

Table 5: All the search-spaces contained in HPO-B-v1

Search Space ID	Eval.	Datasets	Dim.	Search Space Name
151	25	5	1	weka.LMT
5886	295	33	9	weka.J48
5918	192	32	13	weka.IBk
5920	207	29	20	weka.MultilayerPerceptron
5923	265	32	16	weka.RandomForest
5926	83	32	10	weka.SMO_PolyKernel
5978	279	31	17	weka.AttributeSelectedClassifier_J48
6007	248	31	15	weka.AttributeSelectedClassifier_RandomForest
6073	192	32	16	weka.AttributeSelectedClassifier_IBk
6105	256	32	53	weka.FilteredClassifier_RandomForest
6136	300	30	8	weka.FilteredClassifier_J48
6140	186	31	19	weka.FilteredClassifier_IBk
6154	186	31	17	weka.FilteredClassifier-_AttributeSelectedClassifier_IBk
6183	64	32	41	weka.FilteredClassifier_SMO_PolyKernel
6447	270	30	2	weka.FilteredClassifier-_AttributeSelectedClassifier_J48
6458	240	30	3	weka.FilteredClassifier-_AttributeSelectedClassifier_RandomForest
534	302	65	1	weka.Bagging_SMO_PolyKernel
4796	10694	36	3	mlr.classif.rpart.preproc
5499	20	2	12	sklearn.svm.classes.SVC
5988	19	8	31	weka.AttributeSelectedClassifier-_SMO_PolyKernel
5584	12	2	6	sklearn.tree.tree.ExtraTreeClassifier
189	8	2	1	weka.MultiBoostAB_JRip
7064	7	1	8	rm.process.polynomial_by_binomial-_classification_support_vector_machine
5527	385115	51	8	mlr.classif.svm
5636	503439	54	6	mlr.classif.rpart
5859	58809	56	6	mlr.classif.rpart
5860	3100	27	2	mlr.classif.glmnet
5890	1740	58	2	mlr.classif.kknn
5891	44091	51	8	mlr.classif.svm
5906	2289	24	16	mlr.classif.xgboost
5965	414678	60	10	mlr.classif.ranger
5968	918	8	8	mlr.classif.ranger
5970	68300	55	2	mlr.classif.glmnet
5971	44401	52	16	mlr.classif.xgboost
6308	1498	3	2	mlr.classif.glmnet
6762	4746	7	6	mlr.classif.rpart
6766	599056	51	2	mlr.classif.glmnet
6767	491497	52	18	mlr.classif.xgboost
6794	591831	52	10	mlr.classif.ranger
6856	2	1	1	mlr.classif.randomForest
7188	298	35	1	mlr.classif.ranger.imputed.dummied.preproc
7189	2324	40	3	mlr.classif.rpart.imputed.dummied.preproc
7190	1215	32	3	mlr.classif.RRF.imputed.dummied.preproc
7607	18686	58	9	mlr.classif.ranger
7609	41631	59	9	mlr.classif.ranger
5624	2486	5	7	mlr.classif.rpart
124	13	2	1	weka.MultilayerPerceptron

153	5	1	1	weka.Bagging_RandomTree
243	5	1	1	weka.RotationForest_PrincipalComponents_J48
245	5	1	1	weka.RotationForest_PrincipalComponents-_REPTree
246	5	1	1	weka.RotationForest_PrincipalComponents-_RandomTree
247	5	1	1	weka.RotationForest_PrincipalComponents-_RandomForest
248	5	1	1	weka.RotationForest_PrincipalComponents-_LMT
423	10	2	1	weka.AdaBoostM1_SMO_PolyKernel
506	34	9	1	weka.RotationForest_PrincipalComponents_J48
633	5	1	1	weka.RotationForest_PrincipalComponents-_LMT
2553	4	1	1	weka.AttributeSelectedClassifier-_CfsSubsetEval_BestFirst_IBk
4006	8	2	1	weka.LibSVM
5526	149	7	2	mlr.classif.kknn
7286	31	1	2	mlr.classif.glmnet
7290	10	1	4	mlr.classif.rpart
5626	79	3	17	sklearn.ensemble.gradient_boosting-.GradientBoostingClassifier
5889	1433	20	6	mlr.classif.ranger
5458	28	4	16	mlr.classif.develpartykit.ctree
5489	30	5	14	mlr.classif.develpartykit.ctree
4828	264	1	10	mlr.classif.ksvm
214	4	1	1	weka.AdaBoostM1_NaiveBayes
6741	6	1	21	weka.AttributeSelectedClassifier-_MultilayerPerceptron
158	5	1	1	weka.Bagging_JRip
2566	2	1	2	classif.IBk
3894	49	4	6	mlr.classif.rpart.preproc
6003	13537	2	18	mlr.classif.xgboost
6765	1002	3	2	mlr.classif.glmnet
5963	809	2	15	mlr.classif.xgboost
6000	496	1	2	mlr.classif.glmnet
6322	525	2	8	mlr.classif.svm
5623	795	3	3	mlr.classif.glmnet
5969	2543	4	8	mlr.classif.svm
3737	2	1	2	mlr.classif.randomForest
829	4	1	4	weka.Stacking_REPTree
833	5	1	5	weka.Stacking_ZeroR
935	6	1	4	weka.LogitBoost_RandomSubSpace_REPTree
6323	30	1	2	mlr.classif.kknn
7200	811	1	19	mlr.classif.xgboost
673	96	5	1	HIKNN
674	84	5	1	ANHBNN
678	96	5	1	HwKNN
679	96	5	1	KNN
680	83	5	1	NHBNN
681	95	5	1	HFNN
682	95	5	1	DWHFNN
683	95	5	1	CBWkNN
684	95	5	1	NWKNN
685	95	5	1	AKNN

688	11	1	1	HubMiner.HIKNN
689	11	1	1	HubMiner.HwKNN
690	11	1	1	HubMiner.KNN
691	4	1	1	HubMiner.NHBNN
692	11	1	1	HubMiner.HFNN
693	11	1	1	HubMiner.DWHFNN
694	11	1	1	HubMiner.CBWkNN
695	11	1	1	HubMiner.NWKNN
696	11	1	1	HubMiner.AKNN
697	11	1	1	HubMiner.ANHBNN
3994	64	5	1	mlr.classif.kknn.preproc.preproc
5964	506	2	9	mlr.classif.ranger
5972	120	4	2	mlr.classif.kknn
6075	495	1	6	mlr.classif.rpart
2010	7	1	8	weka.SimpleLogistic
2039	26	2	14	weka.AdaBoostM1_RandomForest
2073	133	3	13	weka.Bagging_LogitBoost_DecisionStump
2277	202	3	15	weka.Bagging_RandomForest
3489	141	3	12	weka.NaiveBayes
3490	481	3	29	weka.J48
3502	17	2	6	weka.ZeroR
3960	18	1	15	weka.IterativeClassifierOptimizer_LogitBoost-_DecisionStump
4289	7	1	20	weka.FilteredClassifier_J48
5218	6	1	19	weka.FilteredClassifier_RandomForest
5237	8	1	8	weka.FilteredClassifier_RandomForest
5253	2	1	2	weka.Vote_RandomForest
5295	8	1	13	weka.LogitBoost_DecisionStump
5301	8	1	15	weka.Bagging_LogitBoost_DecisionStump
5315	5	1	9	weka.IBk
2614	6	1	2	matrixnet_on_MagicTelescope_implementation
2629	35	1	6	sklearn.ensemble.forest.RandomForestClassifier
2793	8	1	12	classif.randomForest
2799	7	1	7	sklearn.ensemble.forest.RandomForestClassifier
2823	8	1	11	sklearn.ensemble.forest.RandomForestClassifier
3414	19	1	2	sklearn.ensemble.weight-_boosting.AdaBoostClassifier
3425	8	1	9	sklearn.neural_network-.multilayer_perceptron.MLPClassifier
3434	42	1	19	sklearn.ensemble.forest.ExtraTreesClassifier
3439	6	1	3	sklearn.ensemble.weight-_boosting.AdaBoostClassifier
3442	11	1	5	sklearn.neighbors.classification-.KNeighborsClassifier
5503	14	2	2	sklearn.neighbors.classification-.KNeighborsClassifier
6131	30	1	2	mlr.classif.kknn
5435	2374	4	3	mlr.classif.rpart.preproc
7021	11	1	3	sklearn.naive_bayes.BernoulliNB
5502	6	1	8	sklearn.tree.tree.DecisionTreeClassifier
5521	11	1	3	xgboost.sklearn.XGBClassifier
5604	6	1	12	sklearn.ensemble.forest.ExtraTreesClassifier

5704	11	1	3	DeepForest.DeepForest
5788	3	1	3	__main__.AUC_Booster
5813	51	1	10	__main__.AUCLGBMClassifier
7680	53	1	2	mlr.classif.glmnet
7604	20	1	2	mlr.classif.kknn
5919	6	1	5	weka.SimpleLogistic
5921	7	1	3	weka.Bagging_J48
5922	9	1	10	weka.Bagging_RandomForest
5960	6	1	7	weka.AdaBoostM1_RandomForest
6024	43	1	10	weka.AttributeSelectedClassifier_BayesNet
6124	5	1	18	weka.FilteredClassifier_FilteredClassifier-_RandomForest
6134	6	1	11	weka.J48
6137	7	1	18	weka.FilteredClassifier_NaiveBayes
6139	4	1	16	weka.FilteredClassifier_RandomTree
6155	8	1	7	weka.FilteredClassifier_FilteredClassifier-_FilteredClassifier_IBk
6156	12	1	1	weka.FilteredClassifier_FilteredClassifier_IBk
6182	7	1	3	weka.Bagging_FilteredClassifier_LMT
6189	54	1	20	weka.RandomizableFilteredClassifier-_RandomForest
6190	11	1	6	weka.RandomizableFilteredClassifier-_MultilayerPerceptron
6211	21	1	15	weka.FilteredClassifier_SMO_RBFKernel
6212	20	1	4	weka.RandomizableFilteredClassifier-_SMO_PolyKernel
6213	7	1	3	weka.RandomizableFilteredClassifier-_Bagging_RandomForest
6215	13	1	10	weka.RandomizableFilteredClassifier_J48
6216	8	1	5	weka.RandomizableFilteredClassifier_LMT
6271	14	1	18	weka.FilteredClassifier_Bagging-_AdaBoostM1_J48
6285	19	1	10	weka.FilteredClassifier_FilteredClassifier-_Vote_RandomForest
6309	4	1	7	weka.FilteredClassifier_Bagging_AdaBoostM1-_FilteredClassifier_J48
6345	10	1	20	weka.RandomizableFilteredClassifier-_RandomTree
6347	13	1	2	weka.Bagging_FilteredClassifier_J48
6364	6	1	3	weka.RandomizableFilteredClassifier_BayesNet
6365	10	1	4	weka.RandomizableFilteredClassifier-_NaiveBayes
6376	17	1	7	weka.RandomizableFilteredClassifier-_SimpleLogistic
6433	16	1	18	weka.RandomizableFilteredClassifier-_AdaBoostM1_RandomForest
6461	6	1	8	weka.NaiveBayesUpdateable
6493	14	1	15	weka.FilteredClassifier_FilteredClassifier-_DecisionTable
6507	20	1	3	weka.FilteredClassifier_FilteredClassifier-_RandomizableFilteredClassifier_RandomForest

Table 6: Number of evaluations per search space in the meta-test split of HPO-B-v3

Search Space ID	Dataset ID	No. Evaluations	Space ID	Dataset ID	No. Evaluations
4796	3549	300	5970	49	3164
4796	3918	300	5970	34536	3286
4796	9903	300	5970	14951	3634
4796	23	300	5971	10093	2265
5527	146064	44790	5971	3954	2392
5527	146065	55627	5971	43	3373
5527	9914	59355	5971	34536	3487
5527	145804	59439	5971	9970	3866
5527	31	60574	5971	6566	4254
5527	10101	74531	6766	3903	39155
5636	146064	43841	6766	146064	43785
5636	145804	46496	6766	145953	47439
5636	9914	49668	6766	145804	52737
5636	146065	50509	6766	31	60721
5636	10101	69158	6766	10101	66277
5636	31	79629	6767	146065	44258
5859	9983	2878	6767	145804	45103
5859	31	3020	6767	146064	46066
5859	37	3254	6767	9914	47410
5859	3902	3397	6767	9967	48624
5859	9977	3464	6767	31	68248
5859	125923	5047	6794	145804	52448
5860	14965	200	6794	3	55460
5860	9976	268	6794	146065	60779
5860	3493	389	6794	10101	65952
5891	9889	2584	6794	9914	69200
5891	3899	2587	6794	31	102306
5891	6566	2779	7607	14965	647
5891	9980	2943	7607	145976	677
5891	3891	3083	7607	3896	692
5891	3492	3317	7607	3913	696
5906	9971	254	7607	3903	699
5906	3918	259	7607	9946	722
5965	145836	11228	7607	9967	895
5965	9914	11320	7609	145854	1269
5965	3903	11536	7609	3903	1292
5965	10101	11648	7609	9967	1293
5965	9889	12319	7609	145853	1296
5965	49	12406	7609	34537	1452
5965	9946	13140	7609	125923	1494
5970	37	2829	7609	145878	1593
5970	3492	2947	5889	9971	298
5970	9952	3163	5889	3918	300